

Remarks

The amendments to the claims

The amendments to independent claim 39 is intended to make it clear that the claimed apparatus and methods have to do with transferring a set of database objects "between a medium in which the objects are not organized as a database and a database accessible to the database management system". The amendment is supported at least at page 9, lines 16-24 of the Specification, which sets forth the operations which the claimed apparatus can perform as follows:

- export objects from DBMS 301 in which Data Pump 311 is executing to a set of dump files 323 accessible to DBMS 301;
- export objects from a remote DBMS to which DBMS 301 in which Data Pump 311 is executing has a link to a set of dump files 323 accessible to DBMS 301;
- import objects from a set of dump files accessible to DBMS 301 into DBMS 301;
- import objects from a remote DBMS to which DBMS 301 in which Data Pump 311 is executing has a link into DBMS 301 (termed in the following *fileless import*); and
- make the metadata for a set of database objects in a dump file or a remote DBMS into an SQL script in a file accessible to DBMS 301.

In the above operations, the "medium in which the objects are not organized as a database" is either a set of dump files or the communications medium over which the objects are transferred in fileless import. Amended claim 39 further makes it clear that the steps of the method are "automatically performed by a transfer utility in the database management system". The "transfer utility" is of course embodied in Data Pump 311. Amended claim 41 has been amended to make it clear that the value that specifies the degree of parallelism is "received in the transfer utility". In a preferred embodiment, the value is set using the SET_PARALLEL API, which is described at least at page 18, lines 2-3, page 19, lines 1, 7-10, page 23, lines 2-5, page 29, lines 11-17, and page 37, lines 16-20. As demonstrated by the foregoing, the amendments are completely supported by the Specification as filed.

Traversal of the rejections of claims 1-38 and 43-62

The disclosure of Blasko

In his rejection of claims 1 and 43, Examiner finds the disclosure of the "queryable control database object" in Basko. What Basko discloses is a technique for importing non-database objects contained in files into a database. Basko's FIG. 1, described beginning at col. 3, line 32, gives a good overview. Log data 101 is a file; in a preferred embodiment, it contains click stream data. The items of click stream data are of course not "database objects"; indeed, the whole point of Basko's technique is to make the items of click stream data into "database objects". To do this, he first parses them 102 to produce event data 103, which is the mouse events contained in the click stream, and miscellaneous data 104, which is data associated with the mouse events such as product information and user identifications. The items of event data and miscellaneous data 104 are still not database objects. The information to make these items of data into database objects is contained in import Meta Data table 106, which as a database table:

The import metadata describes how the event and miscellaneous data, referred to collectively as "import data," are to be imported by the import engine and stored in the data warehouse. The import metadata not only defines how the import data is to be transformed, but also defines the structure of the tables of the data warehouse in which the import data is to be stored.

Thus, as shown in the tables in Cols, 4 and 5, the import metadata table defines the table an item of import data is to be written to, the column in the table that the item of data is to be written to, and information specifying how the column is to be defined in SQL.

It should further be pointed out here that Blasko's import metadata table only describes how the items of import data contained in a log data file are to be made into database objects; it does not "specify the set of objects" which are to be added to the database. Blasko's import metadata 106 consequently also cannot "represent the job", as set forth in claims 1 and 43 with regard to the "queryable control database object".

In his rejection of claims 1 and 43, Examiner refers Applicants to col. 10, lines 31-59. What this section of the reference does is describe the tables making up aggregation metadata 109. As explained in overview at col. 3, lines 55-58 of Blasko,

The aggregation metadata defines how the data in the data warehouse is to be aggregated. The aggregation engine inputs the aggregation metadata, aggregates the data of the data warehouse in accordance with the aggregation metadata, and stores the aggregated data.

The details set forth at col. 10, lines 31-59 confirm the overview description. None of this has anything to do with a job that transfers a set of database objects into or out of a DBMS or with a "queryable control database object ... that represents the job and specifies the set of [database] objects" set forth in Applicants' claims 1 and 43.

5

Patentability of claims 1- 35 and 43-65

Claims 1 and 43

As just demonstrated, Blasko does not disclose a "queryable control database object ... that represents the job and specifies the set of [database] objects". Since that is the case, the combined references do not show all of the limitations of claims 1 and 43 and consequently, the combined references cannot render claims 1 and 43 and the claims dependent from those claims obvious.

Claims dependent from claims 1 and 43

The references also do not disclose many of the additional limitations of the claims dependent from claims 1 and 43, and thus these claims are patentable in their own rights over the references. With regard to claims 3-15 and 24-25, Examiner finds all of the added limitations in the Oracle SQL `SELECT` command. The problem here is that all of these limitations concern the control database object. `SELECT` does not disclose a control database object, and consequently cannot disclose these further limitations of the control database object.

With regard to claims 26-35 and claims 50-55 and 58-61, these claims have to do with the manner in which a user interacts with the transfer mechanism. The claims are rejected on the basis of the backup and recovery techniques disclosed in the *Oracle 9i user-managed backup and recovery guide*. Applicants' techniques could of course be used to make a copy of a database, the copy could be used as a backup, and the database could be restored from the backup, but the techniques described in the *backup and recovery guide* are quite different. In brief, what is backed up are the datafiles, which are the files that contain the actual physical information in the database. Additionally, the database maintains redo logs, which are logs of transactions performed on the database. To recover a database, one obtains copies of the backed up datafiles, restores the database from the backed up datafiles, and then applies the data in the redo logs to the restored database until the database is restored to the state it was in when the failure occurred which made the recovery necessary. See in this regard page 1-6 of the *backup*

and recovery guide. As noted at page 1-3 of the *guide*, the datafiles and the redo logs contain the physical information stored in the database; they are not organized as database objects. Consequently, neither making backup datafiles and making redo logs nor restoring a database from backup datafiles and redo logs is a transfer of "a set of database objects into or out of a database system", as set forth in claims 1 and 43. The cited locations in the *backup and recovery guide* further disclose nothing corresponding to the "queryable control database object", and because that is so, the reference cannot be used to reject claims addressed to characteristics of the queryable control database object.

The rejections of claims 16-22, 44-49, and 56-57

These claims are rejected on the basis of the Oracle 9i database documentation, Basko, and Rivlin. Beginning with claims 16-22, the further limitations in these claims all concern the rows in the "queryable control database object" that represent the database objects being transferred. Basko's table represents only metadata; the rows in Rivlin's table represent files, not database objects; the combination of the references thus does not disclose the claimed limitations.

Claims 44-49

These claims set forth one of the advantages of the queryable control database object: because it controls the job and maintains the current state of the job, the job may keep running as users attach to it and detach from it; further, an attached user may determine the current state of the job from the control database object and modify the control database object and thereby modify the manner in which the job is performed. The closest the references come to any of this is Rivlin. There, the file attribute table contains a field for each file that indicates whether the file was successfully transferred to the database. There is, however, no notion in the Oracle documentation's import and export utilities, in Blasko, or in Rivlin of a queryable control database object that makes it possible for the users to attach and detach from a running job.

Claims 56-57

These claims concern the use of the queryable control database object to stop and restart the job; there is also no notion in the Oracle documentation's import and export utilities, in Blasko, or in Rivlin of a queryable control database object that makes it possible for the user to stop and restart a running job.

Claims 62-65

These claims are dependent Beauregard claims and consequently are patentable by virtue of the patentability of the claims they are dependent from.

5 *Traversal of the rejections of claim 36-38*

Claim 36 reads as follows:

36. (previously presented) A set of files for transferring a set of database objects into a database management system,
the set of files comprising:

- 10 at least one file containing the objects belonging to the set thereof; and
 a queryable control database object contained in a file belonging to the set of files that specifies for each object belonging to the set the location of the object in the set of files and an order in which the database management system transfers the object during the transfer.

15

Examiner has rejected this claim under 35 U.S.C. 103 as obvious over the combination of Oracle documentation, Basko, and Rivlin, which was cited in Examiner's first Office action. Examiner does not explicitly indicate what part of the Oracle documentation he is referring to; the *Oracle 9i user-managed backup and recovery guide* states at 1-3,

- 20 User-managed backups can be either logical or physical. You can use the Export utility to make backups of logical objects such as tables, views, and stored procedures, and use the Import utility to restore these objects.

25 The export and import utilities mentioned above are described in Applicants' *Description of related art* at page 3, lines 1-28; as is apparent from the description, there is nothing comparable to the "queryable control database object" in the files produced and consumed by the export and import utilities.

As already pointed out, Blasko's import metadata table is also not the

- 30 queryable control database object contained in a file belonging to the set of files that specifies for each object belonging to the set the location of the object in the set of files and an order in which the database management system transfers the object during the transfer

35 of claim 36.

As regards Rivlin, it is clear from column 3, lines 11-32 and TABLE 1 of Rivlin that Rivlin's file attribute table specifies a set of files, not a set of database objects. Moreover, the file attribute table is not a file in the set of files that is being transferred.

- 5 Since neither the *user-managed backup and recover guide* nor Blasko nor Rivlin discloses Applicants' queryable control database object either individually or in combination, the claim is patentable over the combination of references.

Dependent claims 37 and 38

- 10 These claims are of course patentable because claim 36 is patentable, but they also include additional limitations that do not appear in the references, and are consequently patentable in their own rights over the references. With regard to claim 37, in Blasko, import metadata table 106 is used to process event data 103 and miscellaneous data 104, but Blasko does not process metadata at all. The same is the case with Rivlin. In Oracle import, the order in which the
15 database objects occur in the import file determines the order in which they are processed; there is nothing like the "queryable control database object". As for claim 38, there is simply no disclosure anywhere in the references of the information in the headers of the files in the claimed "set of files".

20 *Patentability of claims 39-42 as amended*

As presently amended, claim 39 reads as follows:

- 25 **39. (currently amended)** A method of transferring database objects between a medium in which the objects are not organized as a database and a database accessible to a database management system, the database management system being capable of employing a plurality of transfer techniques, each database object being associated with metadata that defines the database object's type, the method being automatically performed by a transfer utility in the database management system and
30 the method comprising the steps of:
obtaining metadata for the object type and using the metadata to make a determination of the composition of the objects of the type;
selecting a transfer technique for objects belonging to the type from the plurality thereof according to the determination; and
35 transferring the objects belonging to the type according to the selected transfer technique.

The claims as amended make it clear that the transfer of database objects set forth in the claim is "between a medium in which the objects are not organized as a database and a database". As is well known in the database arts, what is returned by the `SELECT` command is a set of rows. See for example, the description of the `SELECT` command at page 8-2 of the *Oracle 9i rev 2*

5 *SQL reference*:

The list of expressions that appears after the `SELECT` keyword and before the `FROM` clause is called the **select list**. Within the select list, you specify one or more columns in the *set of rows* you want Oracle to return from one or more tables, views, or materialized views. The number of columns, as well as their datatype and length, are determined by the elements of the select list. (emphasis added)

Thus, what `SELECT` returns is a set of row objects that are part of the database upon which the `SELECT` command is executed. Indeed, it is *because* `SELECT` returns row objects that are part of the database that `SELECT` commands may be nested and operations such as unions may be performed on the row objects returned by select commands.

Because what `SELECT` returns is a set of row objects that are part of the database upon which `SELECT` is executed, `SELECT` is not claim 39's method of "transferring database objects between a medium in which the objects are not organized as a database and a database accessible to a database management system" and `SELECT` cannot anticipate claim 39 or any of the claims dependent from claim 39.

Conclusion

Applicants have traversed the rejections of claims 1-38 and 43-65; the amendments to claims 39-42 make explicit what is implicit when the claims are read in the context of the Specification and consequently should not require a new search. Applicants consequently request that Examiner allow claims 1-38 and 43-62 on the basis of the traversals and employ his discretion to enter the amendments to claims 39-42 and allow the claims as amended. Applicants do not believe that this response requires any additional charge. Should there be additional charges, please charge them to deposit account number 501315.

Respectfully submitted,

/Gordon E. Nelson/
Attorney of record,

5

Gordon E. Nelson
57 Central St., P.O. Box 782
Rowley, MA, 01969,
Registration number 30,093
Voice: (978) 948-7632
Fax: (866) 723-0359
1/17/07
Date